

# Anonymizing Data via Polynomial Regression

Jordi Nin<sup>1</sup>

Jordi Pont-Tuset<sup>2</sup>

Pau Medrano-Gracia<sup>2</sup>

Josep L. Larriba-Pey<sup>2</sup>

Victor Muntés-Mulero<sup>2</sup>

<sup>1</sup>IIIA, Artificial Intelligence Research Institute  
CSIC, Spanish National Research Council  
Campus UAB, s/n  
jnin@iiia.csic.es

<sup>2</sup>DAMA-UPC  
Universitat Politècnica de Catalunya  
Campus Nord UPC  
{jpont,pmedrano,larri,vmunes}@ac.upc.edu

## Abstract

The amount of confidential information accessible through the Internet is growing continuously. In this scenario, the improvement of anonymizing methods becomes crucial to avoid revealing sensible information of individuals. Among several protection methods proposed, those based on the use of linear regressions are widely utilized. However, there is not a reason to assume that linear regression is better than using more complex polynomial regressions. In this paper, we present *PoROP-k*, a family of anonymizing methods able to protect a data set using polynomial regressions. We show that *PoROP-k* not only reduces the loss of information, but it also obtains a better level of protection compared to previous proposals based on linear regressions.

## 1 Introduction

Privacy in statistical databases (PSD) [14] and privacy preserving data mining (PPDM) [2] studies the tension between the increasing societal and economical demand for accurate information, and the legal and ethical obligation to protect the privacy of individuals and enterprisers which are the respondents of the statistical data.

Since the use of the Internet has become

very usual in all business areas, privacy is a common concern for all those companies which have sensible data accessible through the web. Also, surveys show that most of the web users are unwilling to provide confidential data into a web site unless privacy protection measures are provided [3].

For this reason, a wide range of anonymizing methods have been proposed. The goal of these methods is to ensure an acceptable level of protection of the confidential data preserving their statistical utility. Good surveys about protection methods can be found in the literature [1, 9].

In [9], the anonymizing methods are classified into two different categories depending on their use of the original values: *synthetic data generators* and *perturbative protection methods*. The synthetic data generators only use original data to build a model and, afterwards, a new data set is built based on this model. The perturbative protection methods are based in the addition of noise into the original data set in order to make it difficult for an intruder to recover the original values.

Linear regression models are commonly used to anonymize data. Two examples of this kind are the *Information Preserving Statistical Obfuscation* (IPSO) [5], a synthetic data generator, and *LiROP-k* methods [11], which include both a set of perturbative protection method and a set of synthetic data genera-

tors, and were developed to solve some drawbacks of IPSO [13]. However, to our knowledge, more complex regression methods have not been presented in the literature.

In this paper, we study a new family of methods called *PoROP-k*, that makes it possible to protect confidential data using more complex regression models. We show in our experiments that incrementing the complexity of the regression model, *PoROP-k* methods outperform LiROP-*k* methods (which are a particular case of the family of methods included in *PoROP-k*), when the *score*, a standard measure to compare protection methods defined in [8], is used to compare both methods.

The structure of the paper is as follows. In Section 2, we present the scenario of our work, in Section 3, we present our protection method using polynomial regression. Then, in Section 4 we describe the experiments. Finally, the paper draws some conclusions and a description of future work.

## 2 Privacy Protection Scenario

Before presenting our proposal, we first present the protection scenario assumed in this work.

The main objective of a protection method is to anonymize a data set. A data set can be viewed as a file containing a number of records, where each record contains a set of attributes of an individual. The attributes in the original data set can be classified into two different categories, depending on their capability to identify unique individuals, as follows:

- **Identifiers.** The identifier attributes are used to identify the individual unambiguously. A typical example of identifier is the passport number.
- **Quasi-identifiers.** A quasi-identifier attribute is an attribute that is not able to identify a single individual when it is used alone. However, when it is combined with other quasi-identifier attributes, they can uniquely identify an individual. Among

the quasi-identifier attributes, we distinguish between confidential and non-confidential, depending on whether they contain confidential information. An example of non-confidential quasi-identifier attribute would be the postal code, while a confidential quasi-identifier might be the salary.

When a data set is protected, identifiers are removed or encrypted to prevent an intruder to re-identify individuals easily. Typically, the remaining attributes are released, some of them protected. In this paper, we assume that non-confidential attributes are protected, while confidential attributes are not. This allows third parties to have precise information on confidential data without revealing to whom that confidential data belongs to.

In this scenario, as shown in Figure 1, an intruder might try to re-identify individuals by obtaining the non-confidential quasi-identifier data ( $Y$ ) together with identifiers ( $Id$ ) from other data sources. Applying record linkage between the protected attributes ( $Y'$ ) and the same attributes obtained from other data sources ( $Y$ ), the intruder might be able to re-identify a percentage of the protected individuals together with their confidential data ( $X$ ). This is what protection methods try to prevent.

## 3 Method description

Analogously to LiROP-*k* methods, polynomial regression on Ordered Partitions (PoROP-*k*) methods pre-process the original data using three basics steps: vectorization, sorting and partitioning. There are several aspects that motivate these three steps:

**Vectorization.** The main idea of this first step is to gather all the values in the data set in a single vector, independently of the attribute they belong to. Consequently, we are ignoring the attribute semantics and, therefore, all the possible relationships, like covariance or correlations among the attributes in the data set.

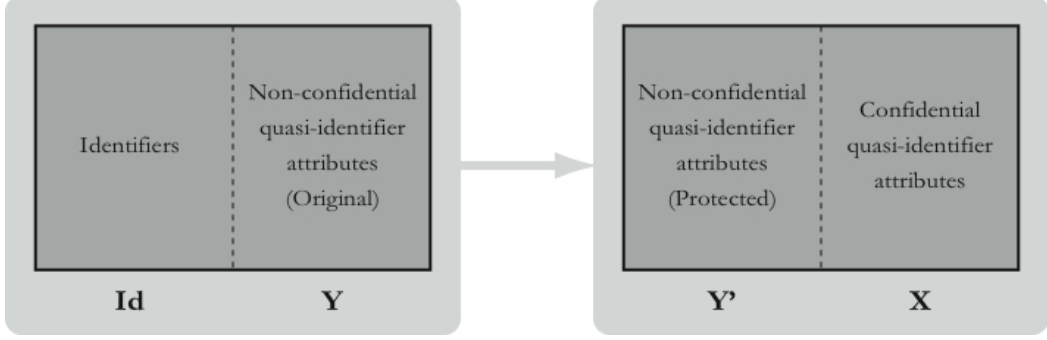


Figure 1: Re-identification scenario.

**Sorting.** The second step is to sort all the vectorized values. This step is necessary in order to fit the data into a model in a easy way. Note that sorting the values is a way of adding noise.

**Partitioning.** Even taking into account that data is sorted, using a unique model to fit all the data is unfeasible because the error of the model could be very large. In order to improve the accuracy, the sorted vectorized data is split into several  $k$ -partitions. Then a different model regression is used to fit the data of each partition. Modifying the value of  $k$ , PoROP- $k$  methods allow us to tune the accuracy of the regression model by changing the size of the partition being fitted. Note that if the data set was not sorted,  $k$  would not have this property.

Since the range of the values in the different attributes could differ significantly among them, it might happen that the sorting step does not merge all the attributes appropriately. For this reason, it is necessary to normalize the data. There are many ways to normalize a data set. A possible solution would be to normalize each attribute independently before the application of the vectorization step.

This normalization method could present problems with skewed attributes and, therefore, the attributes could not be merged in the sorting step. For this reason, we propose

to normalize the data stored in each partition independently. This way, similar values are put in the same partition and, therefore, the chances to avoid the effect of skewness in the data is higher. Once the data is normalized, vectorization, sorting and partitioning steps are repeated.

Formally speaking, let  $\mathcal{D}$  be the original data set to be protected. We denote by  $R$  the number of records in  $\mathcal{D}$ . Each record consists of  $a$  numerical attributes or fields. We assume that none of the registers contain blanks. We denote by  $N$  the total number of values in  $\mathcal{D}$ . As a consequence,  $N = R \cdot a$ .

Let  $V$  be a vector of size  $N$ . First,  $V$  is sorted increasingly. Let us denote by  $V_s$  the ordered vector of size  $N$  containing the sorted data and  $v_i$  the  $i$ th element of vector  $V_s$ , where  $0 \leq i < N$ .

Next,  $V_s$  is divided into smaller sub-vectors or partitions. Then, each sub-vector is normalized into the  $[0, 1]$  interval and they are all sorted and partitioned again. We define  $k$ , where  $1 < k \leq N$ , as the number of values per partition. Note that, if  $k$  is not a divisor of  $N$  the last partition will contain a smaller number of values. Let  $P$  be the number of  $k$ -partitions. We call  $r$  the number of values in the last partition where  $0 \leq r < k$ . Therefore,  $N = kP + r$ . If  $r > 0$ , we have  $P + 1$  partitions. We denote by  $P_m$  the  $m$ th partition.

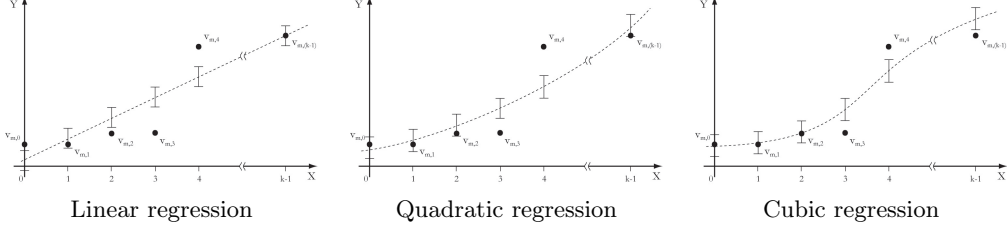


Figure 2: An example of a set of points from a partition, its model regression and the more probable interval for the protected value when noise is added independently.

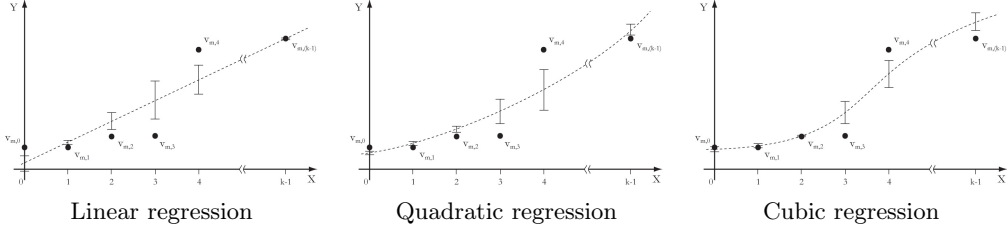


Figure 3: An example of a set of points from a partition, its model regression and the more probable interval for the protected value when noise is added taking into account the original value.

Let  $v_{m,n}$  be defined as the  $n$ th element of  $P_m$ :

$$\begin{cases} v_{m,n} := v_{mk+n} & n = 0 \dots k-1 \quad m = 0 \dots P-1 \\ v_{P,n} := v_{Pk+n} & n = 0 \dots r-1 \end{cases}$$

For each  $P_m$ , a regression model is computed over the following  $(X, Y)$  points:

$$(0, v_{m,0}) \quad (1, v_{m,1}) \quad \dots \quad (k-1, v_{m,(k-1)})$$

When  $r > 0$ , the size of the last partition ( $P_P$ ) is  $r < k$ . In this case, the regression model of this partition is computed differently: the nearest last  $k$  points of the data set are used to compute the regression model, but only the  $r$  points held by  $P_P$  are actually protected. This guarantees that each regression model is computed using the same number of points, so the level of accuracy is homogeneous. Therefore, in this case, the fitting for the last partition is computed over the following  $(X, Y)$  points:

$$(0, v_{m,N-k}) \quad (1, v_{m,N-k+1}) \quad \dots \quad (k-1, v_{m,N-1})$$

Finally, when the regression model is computed, PoROP- $k$  methods add Gaussian noise

to the polynomial regression to partially change the order of the points. With the addition of noise, it will be more difficult for an intruder to reveal the original data even knowing the values of some attributes.

Similarly to LiROP- $k$  methods, PoROP- $k$  methods can be considered both a protection method and a synthetic data generator depending on the way used to add noise. If the Gaussian noise is computed independently of the original value to protect, PoROP- $k$  methods can be considered synthetic data generators. We call this configuration PoROP- $s-k$  and is described in Figure 2. On the other hand, if the noise addition is dependent on the point to be protected, PoROP- $k$  methods must be considered perturbative. In this latter case, we call this configuration PoROP- $p-k$ . An example of these is presented in Figure 3. More details about noise addition methods based on linear regression models can be found in [11].

Following, we present the formulas to compute PoROP- $k$  methods using linear and quadratic regressions. Formulas for cubic regressions, although they are used later in the

experiments, are omitted in this paper due to their length and complexity.

### 3.1 PoROP- $k$ using linear regression

When a linear regression is used to model the data in each partition, PoROP- $k$  methods become LiROP- $k$  methods, since these last subset is a particular case of our proposal. Assuming that the resulting linear regression which models the data is  $l_{m,n} = \alpha_m n + \beta_m$  (where  $n = 0 \dots k-1$ ), then the expressions used to compute  $\alpha_m$  and  $\beta_m$  are as follows:

$$\alpha_m = \frac{2}{k(k+1)} \left[ (2k-1) \sum_{n=0}^{k-1} v_{m,n} - 3 \sum_{n=1}^{k-1} n v_{m,n} \right]$$

$$\beta_m = \frac{2}{k(k+1)} \left[ -3 \sum_{n=0}^{k-1} v_{m,n} + \frac{6}{k-1} \sum_{n=1}^{k-1} n v_{m,n} \right]$$

These results can be derived from the normal equations as presented in [6].

### 3.2 PoROP- $k$ using quadratic regression

However, as mentioned previously, PoROP- $k$  methods allow to use more complex models. In this subsection we present the equations used to build a quadratic model, assuming that the resulting quadratic regression is  $l_{m,n} = \alpha_m n^2 + \beta_m n + \gamma_m$  (where  $n = 0 \dots k-1$ ). Specifically, the expressions used to compute  $\alpha_m$ ,  $\beta_m$  and  $\gamma_m$  are presented in Figure 4. Analogously to the linear regression, these results can be derived from the normal equations.

## 4 Experiments

In Section 3, we have presented the PoROP- $k$  protection methods, which protect a data set combining a new vision of the data to be protected with a complex pre-processing process and a model regression. In this section, we describe a set of experiments that allow us to test the new set of methods presented in this paper and compare them to the more simple linear models.

### 4.1 Data

For evaluation purposes, we have considered the two reference data sets proposed in the CASC project [4]. The first has been extracted using the Data Extraction System (DES) from the U. S. Census Bureau [7], called Census. The second has been obtained from the U.S. Energy Information Authority [10], called EIA.

The Census data set contains 1080 records consisting of 13 attributes (which is equal to 14040 values to be protected). The EIA data set, after removing the identifiers and the categorical attributes, contains 4092 records consisting of 5 attributes. The total number of values to be protected in this data set is equal to 20460.

### 4.2 Measures

In order to evaluate PoROP- $k$  methods we calculate the *score*, a typical general measure used to compare different protection methods [9]. We have used this score to compare PoROP- $k$  methods with LiROP- $k$  methods.

In order to calculate the *score*, we use the measures presented in previous work:

- **Information Loss (IL):** Let  $X$  and  $X'$  be matrices representing the original and the protected data set, respectively. Let  $V$  and  $R$  be the covariance matrix and the correlation matrix of  $X$ , respectively; let  $\bar{X}$  be the vector of variable averages for  $X$  and let  $S$  be the diagonal of  $V$ . Define  $V'$ ,  $R'$ ,  $\bar{X}'$ , and  $S'$  analogously from  $X'$ . The information loss is computed by averaging the mean variations of  $X - X'$ ,  $V - V'$ ,  $S - S'$ , and the mean absolute error of  $R - R'$  and multiplying the resulting average by 100. All these measures have been extracted from [9] and are computed in the same way.
- **Disclosure Risk (DR):** We use the three different methods presented in [12] in order to evaluate DR: (i) *Distance Linkage Disclosure risk* (DLD), which is the average percentage of linked records

$$\begin{aligned}
\alpha_m &= 3 \frac{(3N^2 - 3N + 2) \sum_{k=0}^{N-1} p_k}{N(N^2 + 3N + 2)} - 18 \frac{(2N - 1) \sum_{k=0}^{N-1} kp_k}{N(N^2 + 3N + 2)} + 30 \frac{\sum_{k=0}^{N-1} k^2 p_k}{N(N^2 + 3N + 2)} \\
\beta_m &= -18 \frac{(2N - 1) \sum_{k=0}^{N-1} p_k}{N(N^2 + 3N + 2)} + 12 \frac{(16N^2 - 30N + 11) \sum_{k=0}^{N-1} kp_k}{N(N^4 - 5N^2 + 4)} - 180 \frac{\sum_{k=0}^{N-1} k^2 p_k}{N(N^3 + N^2 - 4N - 4)} \\
\gamma_m &= 30 \frac{\sum_{k=0}^{N-1} p_k}{N(N^2 + 3N + 2)} - 180 \frac{\sum_{k=0}^{N-1} kp_k}{N(N^3 + N^2 - 4N - 4)} + 180 \frac{\sum_{k=0}^{N-1} k^2 p_k}{N(N^4 - 5N^2 + 4)}
\end{aligned}$$

Figure 4: Equations to model a data set using linear regression.

using distance based record linkage, (ii) *Probabilistic Linkage Disclosure risk* (PLD), which is the average percentage of linked records using probabilistic based record linkage and (iii) *Interval Disclosure risk* (ID) which is the average percentage of original values falling into the intervals around their corresponding masked values. The three values are computed over the number of attributes that the intruder is assumed to know that, in our case, ranges from one to half of the attributes. These measures have been extracted from [9] and are computed in the same way:

$$DR = 0.25 DLD + 0.25 PLD + 0.5 ID$$

- **Score:** A final score measure is computed by weighting the presented measures, also proposed in [9]:

$$score = 0.5 IL + 0.5 DR$$

### 4.3 Results

In order to understand whether using more complex regression methods allows us to preserve the information more accurately, we first study the information loss of each method.

We test PoROP<sub>s</sub>-*k* and PoROP<sub>p</sub>-*k* methods using linear, quadratic and cubic regressions. The range of values for the number of points per partition *k* has been defined in order to

<i>Census</i>			
<i>k</i>	<i>Linear</i>	<i>Quadratic</i>	<i>Cubic</i>
2000	0.3	0.1	0.1
5000	3.9	0.9	0.6
6000	16.2	1.9	1.7
7000	42.3	12.1	3.4
10000	99.3	33.9	24.1

Table 1: Average result of IL for the PoROP<sub>p</sub>-*k* methods using the Census data set.

<i>EIA</i>			
<i>k</i>	<i>Linear</i>	<i>Quadratic</i>	<i>Cubic</i>
6000	17.9	15.4	14.5
10000	25.9	18.1	17.4
11000	64.8	23.6	17.6
12000	56.0	26.7	21.9
180000	119.7	52.0	33.3

Table 2: Average result of IL for the PoROP<sub>p</sub>-*k* methods using the EIA data set.

make the IL range between 0 to 100. For this reason, *k* values are different in each data set.

We have executed each configuration ten times performing 200 tests in total. The average IL for each configuration is presented in Tables 1 and 2. Tables 3 and 4 show the average scores obtained from the experiments. Note that the tables presented in this section only show the results using the perturbative version (PoROP<sub>p</sub>-*k*). The results obtained by

PoROP<sub>s</sub>- $k$  are almost identical and are omitted for the sake of simplicity.

In most cases, being able to control the IL is very interesting, specially when keeping the statistic in the protected data set is important. As we can see in the tables, PoROP- $k$  methods can control the IL by modifying parameter  $k$ . Usually, when parameter  $k$  increases, IL increases. Note that this happens independently of the model regression and the data set. In our case, the pre-processing phase is very important to guarantee a strong correlation between  $k$  and IL, since by vectorizing, ordering, partitioning and normalizing we make possible to find a regression model that accurately fits the data set.

Observing the same tables, we can see that the more complex is the polynomial model, the lower is the information loss. This happens because by increasing the complexity of the regression function, we also increase the fitting capabilities of the complex polynomial models.

If we observe the score values shown in Tables 3 and 4, we can see that, increasing the complexity of the regression functions, we achieve better quality in the protection. Specifically, in the Census data set, the best scores are obtained using quadratic regression (32.9), while the best scores using linear regressions are 39.7. Analogously, using the EIA data set the best scores are obtained using cubic regressions instead of linear regressions.

The reduction in the score values happens since the reduction of the information loss is larger than the increase of the disclosure risk, compared to previous techniques based on regression models. Note that, as we have explained in Section 4.2 both measures are weighted equally.

## 5 Conclusions and future work

In this paper we have presented a generalization of the LiROP- $k$  anonymizing methods, which we have called PoROP- $k$  methods. We have shown that, by increasing the complexity of the regression model used to protect data, the information loss is reduced and the overall quality of the protection method increases.

<i>Census</i>			
$k$	<i>Linear</i>	<i>Quadratic</i>	<i>Cubic</i>
2000	42.1	37.0	37.0
5000	37.7	41.1	41.6
6000	42.0	41.4	36.9
7000	39.7	32.9	35.6
10000	63.4	41.1	42.9

Table 3: Average scores for the PoROP<sub>p</sub>- $k$  methods using the Census data set.

<i>EIA</i>			
$k$	<i>Linear</i>	<i>Quadratic</i>	<i>Cubic</i>
6000	36.5	37.9	38.5
10000	32.9	33.3	33.7
11000	48.1	38.8	33.6
12000	43.3	33.1	33.9
180000	69.9	38.4	32.3

Table 4: Average scores for the PoROP<sub>p</sub>- $k$  methods using the EIA data set.

Also, our new class of methods allows us to control the information loss by modifying a parameter.

As future work, we plan to find new criteria to decide which is the best regression model for each partition in order to minimize the information loss preserving the disclosure risk as low as possible.

## Acknowledgments

The authors want to thank Generalitat de Catalunya for its support through grant number GRE-00352 and Ministerio de Educación y Ciencia of Spain for its support through grant TIN2006-15536-C02-02. Jordi Nin wants to thank the Spanish Council for Scientific Research (CSIC) for his I3P grant.

## References

- [1] Adam, N. R., Wortmann, J. C., (1989), Security-Control for statistical databases: a comparative study. ACM Computing Surveys, Volume: 21, 515-556.

- [2] Agrawal, R., Srikant, R., (2000), Privacy Preserving Data Mining, Proc. of the ACM SIGMOD Conference on Management of Data, Pages: 439–450.
- [3] Ackerman, M., Faith Cranor, L., Reagle, J., (1999), Privacy in e-commerce: examining user scenarios and privacy preferences, EC '99: Proceedings of the 1st ACM conference on Electronic commerce, ACM Press, ISBN: 1-58113-176-3, Pages: 1–8.
- [4] Brand, R., Domingo-Ferrer, J., and Mateo-Sanz, J. M., (2002) Reference datasets to test and compare sdc methods for protection of numerical microdata. European Project IST-2000-25069 CASC, <http://neon.vb.cbs.nl/casc>.
- [5] Burrige, J., (2003), Information preserving statistical obfuscation. Statistics and Computing, Volume: 13, 321-327.
- [6] Dahlquist, G., Björck, A., (2003), Numerical methods, Mineola, Dover Publications
- [7] Data Extraction System, U.S. Census Bureau, <http://www.census.gov/>
- [8] Domingo-Ferrer, J., Torra, V., (2001), Disclosure Control Methods and Information Loss for Microdata, Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, Elsevier Science, 91-110.
- [9] Domingo-Ferrer, J., Torra, V., (2001), A Quantitative Comparison of Disclosure Control Methods for Microdata, Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, Elsevier Science, 111-133.
- [10] U.S. Energy Information Authority, <http://www.eia.doe.gov/>
- [11] Medrano-Garcia, P., Pont-Tuset, J., Nin, J., Muntés-Mulero, V., (2007), Ordered Data Set Vectorization for Linear Regression on Data Privacy, Lecture Notes in Artificial Intelligence, Springer, volume 4617 pages: 361-372, 12 pages. This work will be presented in the Intl. Conf. on Modelling Decisions for Artificial Intelligence (acceptance rate, 21.7%)
- [12] Torra, V., Domingo-Ferrer, J., (2003), Record linkage methods for multi-database data mining, Information Fusion in Data Mining, Springer, 101-132.
- [13] Torra, V., Abowd, J. M., Domingo-Ferrer, J., (2006), Using Mahalanobis Distance-Based Record Linkage for Disclosure Risk Assessment, Lecture Notes in Computer Science, Volume: 4302, 233-242.
- [14] Willenborg, L., De Waal, T., (2001), Elements of Statistical Disclosure Control, Lecture Notes in Statistics, Springer.